

### 1. Wybrane algorytmy sortowania

- 1.1. Sortowanie przez wybór
- 1.2. Sortowanie bąbelkowe
- 1.3. Sortowanie pozycyjne

### 2. Najważniejsze własności algorytmów

### 3. Poprawność algorytmów

### 4. Skończoność algorytmów

### 5. Złożoność obliczeniowa algorytmów

- 5.1. Złożoność czasowa
- 5.2. Złożoność pamięciowa

### 6. Efektywność algorytmów



#### Warto powtórzyć

1. Przypomnij algorytm szukania najmniejszego elementu wśród  $n$  liczb.
2. W jakich programach spotkaliśmy się z porządkowaniem danych?
3. Jakie znasz sposoby porządkowania elementów dowolnych zbiorów?
4. Jaka jest zależność między algorytmem a programem komputerowym (temat C4, *Informatyka podstawowa*)?
5. Co to jest specyfikacja zadania (problemu)?
6. Przypomnij iteracyjną definicję silni.
7. Na czym polega rekurencja?
8. Podaj przykłady zastosowań stosu.
9. Omów rekurencyjną realizację algorytmu obliczania silni.

## 1. Wybrane algorytmy sortowania

Zarówno człowiek, jak i komputer pewne operacje wykonują szybciej na uporządkowanych zbiorach elementów.

Przetwarzanie danych w bazach danych oparte jest głównie na danych uporządkowanych, np. na posortowanej liście nazwisk sprawniej odnajduje się osobę o konkretnym nazwisku.

Często sortujemy także listy plików, np. według nazw czy dat utworzenia.

W życiu codziennym porządkujemy różne zbiory elementów, nierzadko nie zastanawiając się nawet nad tym, w jaki sposób wykonujemy tę czynność. W algorytmice algorytmy sortowania bardzo precyzyjnie nazwano, usystematyzowano i określono. W tym temacie przedstawimy trzy rodzaje algorytmów sortowania: sortowanie bąbelkowe, kubitkowe i przez wybór – inne omówimy w dalszej części podręcznika.

**Sortowanie** (ang. *sorting*)  
– porządkowanie informacji  
według określonego  
kryterium.